인공지능 보안

-04-

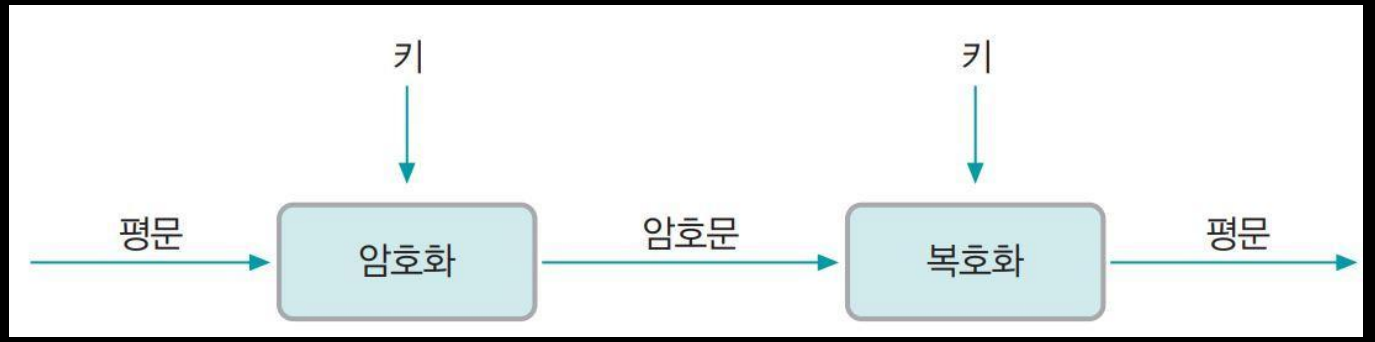
암호의 이해

암호의 개념과 원리

암호화와 복호화

- 암호는 암호문이 노출되더라도 정보를 숨길 수 있어야 함

- 암호화: 평문을 암호문으로 바꾸는 것
- 복호화: 암호문을 평문으로 바꾸는 것
- 암호화 알고리즘: 암호화나 복호화를 수행할 때 양쪽이 알고 있어야 할 수단
- 암호화 키: 약속한 규칙



암호화 방식

- 전치법
  - 단순히 메시지에 들어 있는 문자 위치를 바꾸는 방법
  - 미리 정해둔 문자 배열 규칙으로 암호화와 복호화 수행
  - 스파르타에서 군사용으로 사용하던 봉 암호화

- 대체법
  - 메시지의 글자를 다른 글자로 대체하여 암호화하는 방법
  - 적절한 배합을 찾으면 쉽게 복호화할 수 있는 전치법의 문제를 해결하기 위해 등장
  - 단일 치환과 다중 치환으로 나눌 수 있음

단일 치환 암호화

- 시저 암호화
  - 알파벳 스물여섯 자를 세 자 또는 네 자씩 오른쪽으로 이동한 뒤 해당되는 글자로 변환하여 암호화하는 것
  - 500년 동안이나 사용된 방법이지만, 암호화가 가능한 경우의 수가 26에 불과한 매우 취약한 방식

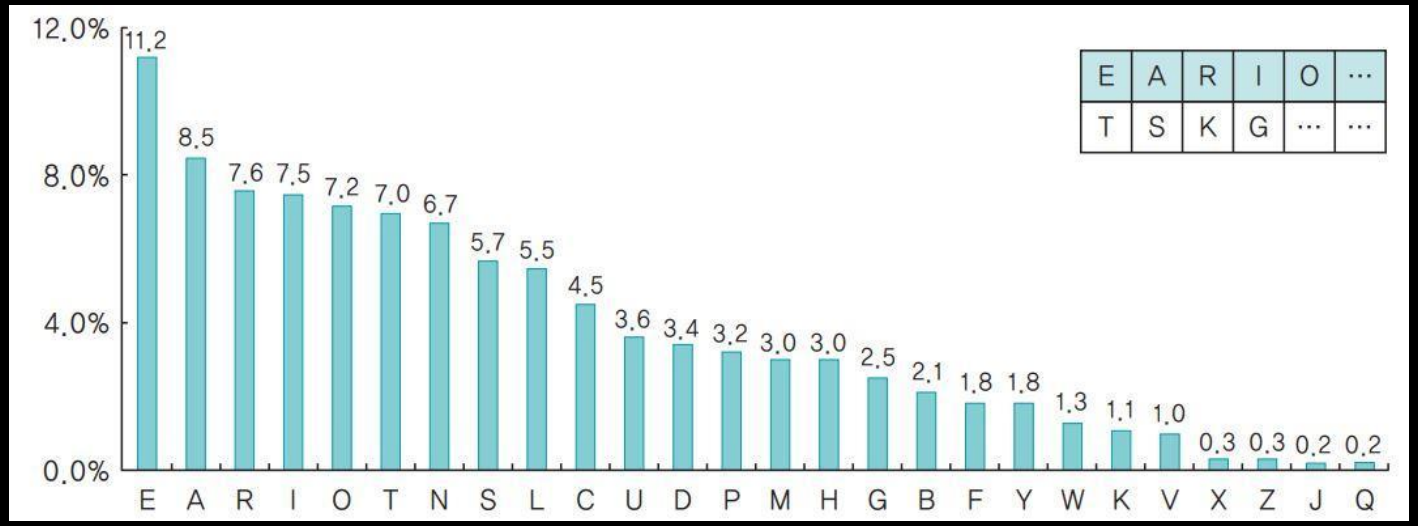
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
x	y	z	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w

- 모노 알파베틱 암호화
  - 알파벳 스물여섯 자를 각각 다른 알파벳에 대응시켜 알파벳을 암호화하는 것
  - 모노 알파베틱 암호문을 복호화하려면 알파벳 대칭표가 있어야 함

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
A	S	I	N	T	O	R	U	V	W	X	Y	Z	B	C	D	E	F	G	H	J	K	L	M	N	P

단일 치환 암호화

- 단일 치환 암호법은 키워드를 몰라도 복호화가 가능
- 빈도 분석법: 알파벳 스물여섯 자가 문장에서 비슷한 빈도로 사용된다는 통계에서 착안한 것
- 단일 치환 암호법의 암호문에 사용된 알파벳의 빈도를 계산해서 통계와 비교해 알파벳을 확인할 수 있음



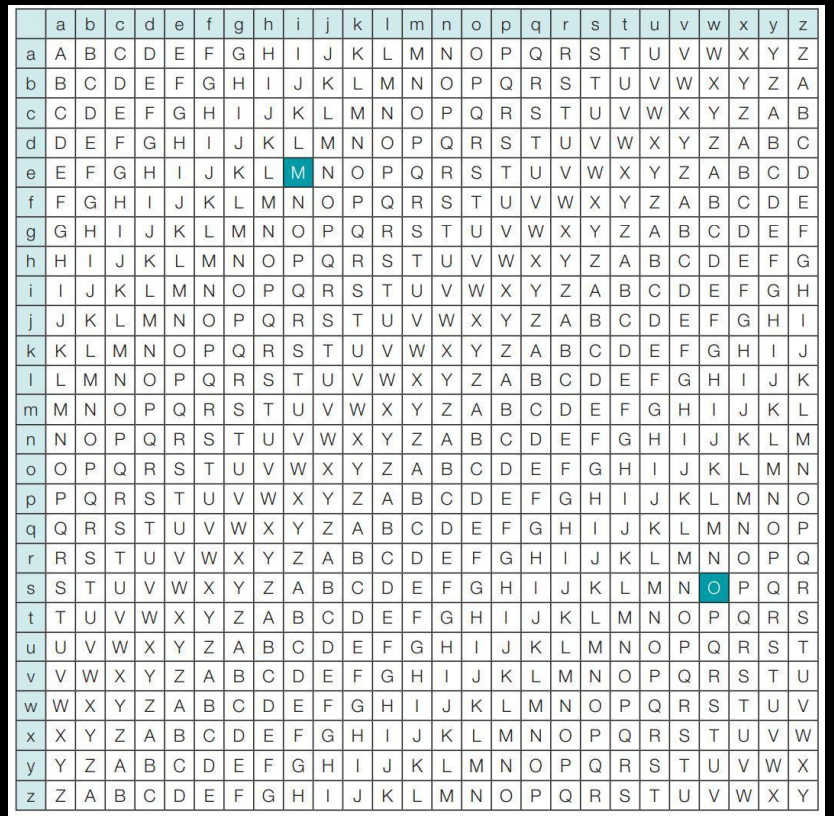
다중 치환 암호화

- 암호화 키와의 매핑에 따라 알파벳 하나가 여러 가지 다른 알파벳으로 대체되어 암호화되는 것

- 비즈네르 암호화
  - 26×26의 알파벳 대칭표를 이용하여 암호화하고자 하는 평문과 암호화 키를 매핑하고 암호화와 복호화를 수행하는 방식
  - 16세기에 프랑스 외교관 블레즈 비즈네르가 만듬

다중 치환 암호화

- 암호화 과정
  - 암호화하려는 평문: ‘wish to be free from myself’
  - 암호화 키: ‘secret is beautiful’
  - 평문의 첫 문자인 w를 비즈네르 표의 가로축에서 찾고 암호화 키의 첫 문자인 s를 세로축 에서 찾으면 O에 대칭
  - 평문의 두 번째 문자 i와 암호화 키의 두 번째 문자 e를 비즈네르 표에서 찾으면 M에 대칭
  - 평문 ‘wish to be free from myself’는 ‘OMUY XH JW GVEY YZTG XQWGCJ’ 라는 암호문
- 비즈네르 복호화 과정
  - 암호화 키의 첫 문자인 s를 비즈네르표의 가로축에서 찾고
  - 평문의 첫 문자인 w를 세로축에서 찾음



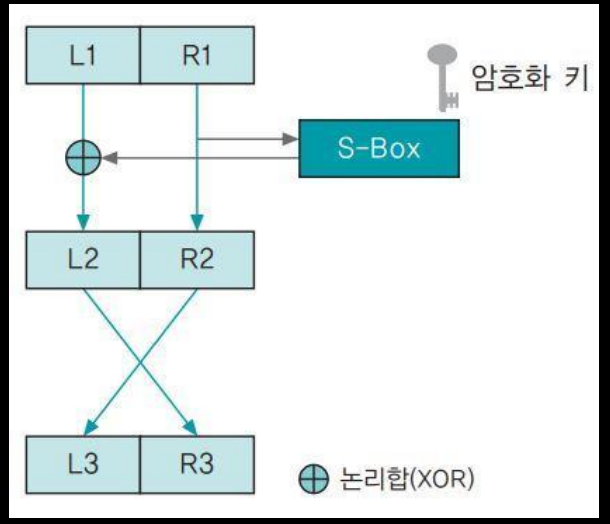
	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
b	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
k	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

그림 7-6 비즈네르 표

대칭키 암호화

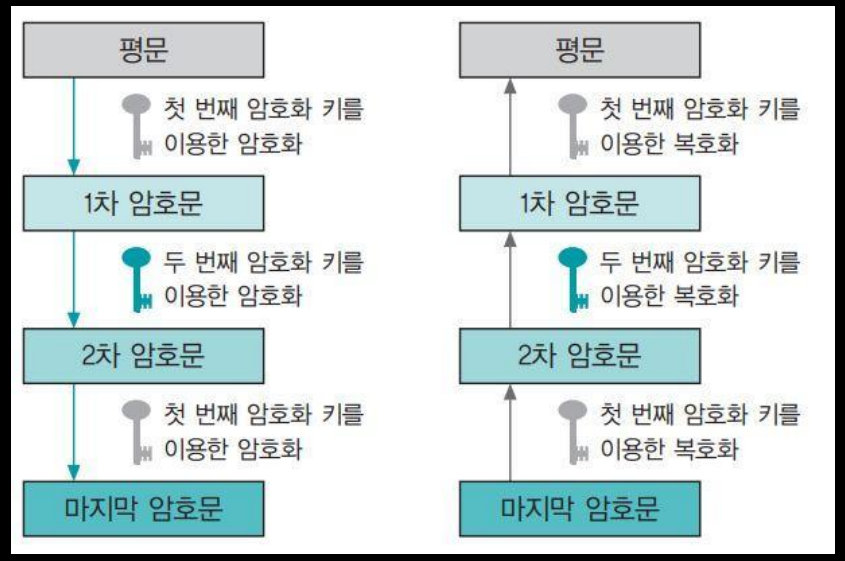
DES(Data Encryption Standard)

- 1972년 미국 상무부의 NBS에서 정보 보호를 목적으로 공모한 암호화 방식
- IBM의 바터 투흐만과 칼 마이어가 개발
- 1977년 1월 NIST에 의해 암호화 표준으로 결정
- DES는 64비트의 블록 암호화 방식으로 56비트 크기의 암호화 키로 암호화
- 미국 정부는 1998년부터 DES 사용을 중단했지만 아직도 여러 응용 프로그램에서 많이 사용



Triple DES

- DES의 복호화가 가능해짐에 따라 AES가 나오기 전까지 임시로 사용
- 암호화 및 복호화 과정에서 2개의 암호화 키를 이용
- DES 알고리즘보다 암호화 강도가 2배 더 높아 오래 사용되지 못함



AES(Advanced Encryption Standard)

- DES의 암호화 강도가 점점 약해지면서 새롭게 개발된 것
- 1997년에 NIST는 암호화 방식을 다시 공모
- 향후 30년 정도 사용할 수 있는 보안성, 128비트 암호화 블록, 다양한 키 길이를 갖출 것이라는 공모 조건
- 빈센트 레이먼, 요안 다에먼이 개발한 Rijndael 알고리즘이 2000년 10월 최종 AES 알고리즘으로 선정

국내 대칭키 암호화 방식

- SEED
  - 전자상거래, 금융, 무선통신 등에서 전송되는 중요한 정보를 보호하기 위해 한국인터넷진흥원과 국내 암호 전문가들이 순수 국내 기술로 개발한 128비트 블록의 암호화 알고리즘
  - SEED 128은 1999년 9월 정보통신단체표준(TTA 표준)으로 제정되었고, 2005년에는 ISO/IEC와 IETF로부터 암호화 표준 알고리즘으로 인정
  - 국내에서 개발된 많은 암호 프로그램과 보안 솔루션에서 사용
- ARIA
  - 전자정부 구현으로 다양한 환경에 적합한 암호화 알고리즘이 필요하여 국가보안기술연구소(NSRI) 주도로 개발
  - 2004년 국가표준기본법에 의거하여 국가표준Ⓚ으로 지정
  - AES 알고리즘과 마찬가지로 128/192/256비트 암호화 키를 지원

비대칭키(공개키) 암호화

Chapter 8
Security

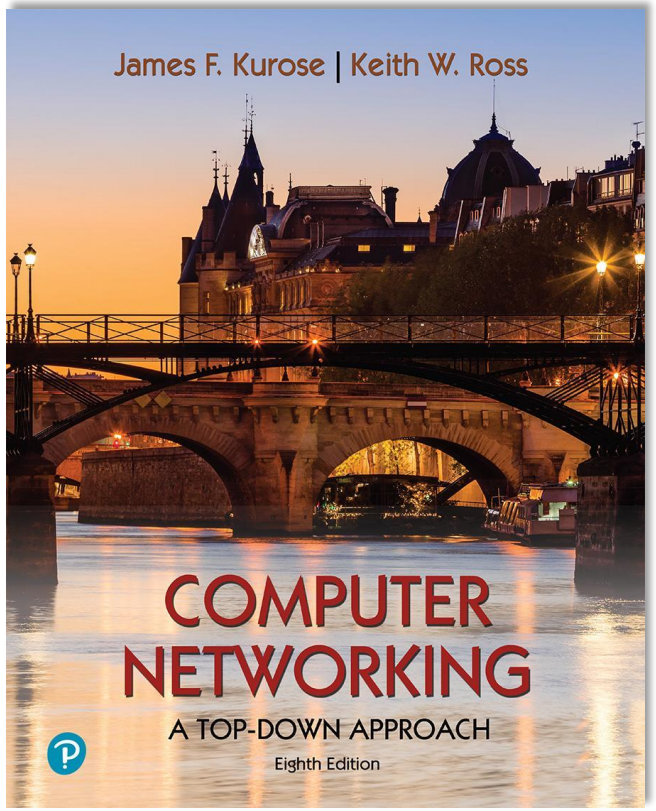
A note on the use of these PowerPoint slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you see the animations; and can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- If you use these slides (e.g., in a class) that you mention their source (after all, we'd like people to use our book!)
- If you post any slides on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

For a revision history, see the slide note for this page.

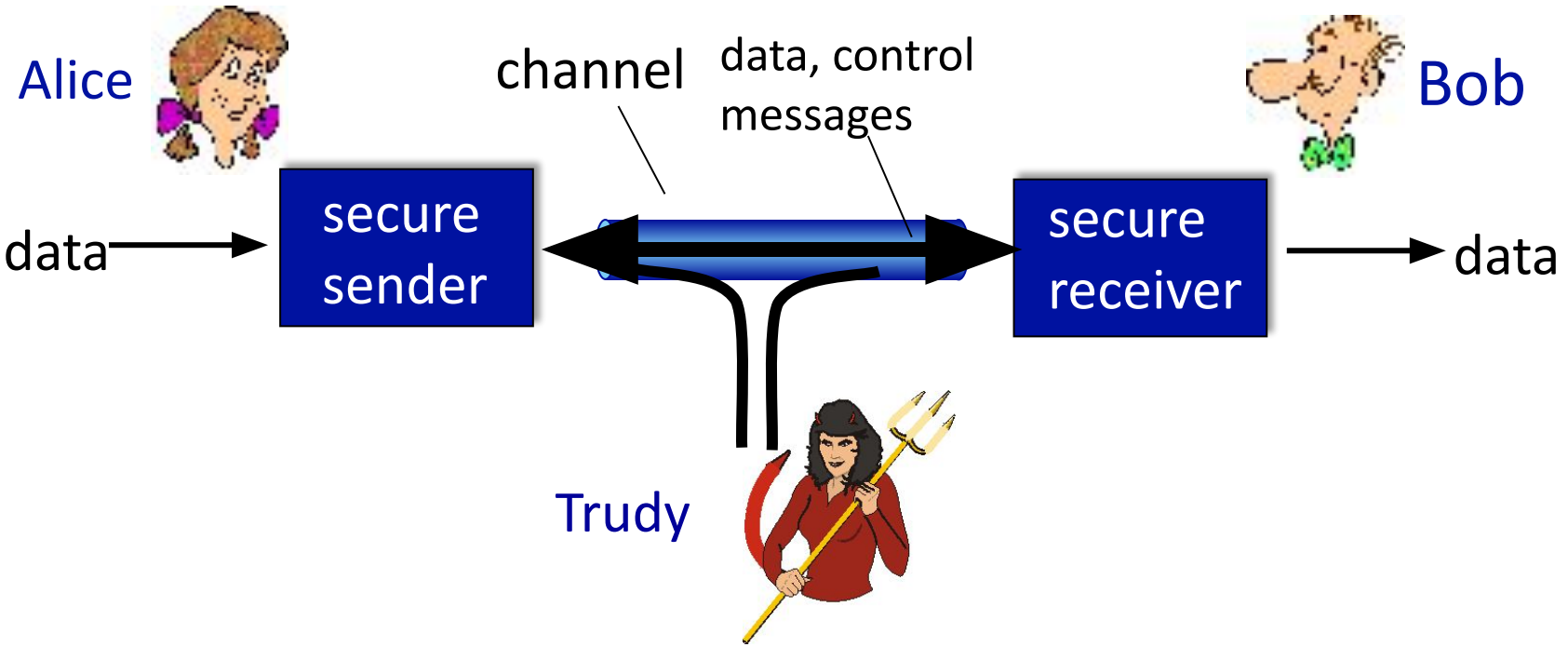
Thanks and enjoy! JFK/KWR



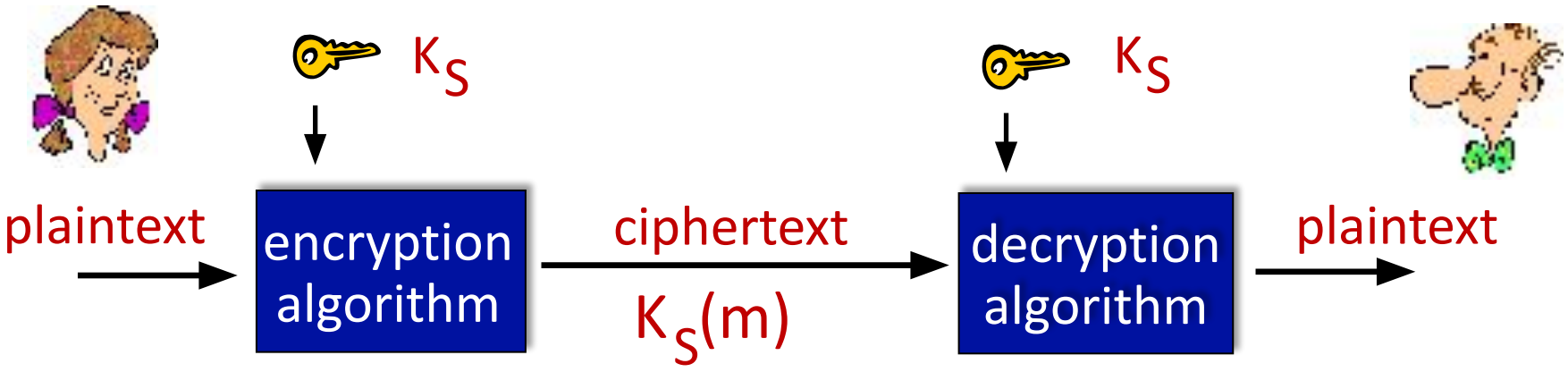
Computer Networking: A Top-Down Approach
8th edition
Jim Kurose, Keith Ross
Pearson, 2020

Friends and enemies: Alice, Bob, Trudy

- well-known in network security world
- Bob, Alice (lovers!) want to communicate “securely”
- Trudy (intruder) may intercept, delete, add messages



Symmetric key cryptography



symmetric key crypto: Bob and Alice share same (symmetric) key: K

- *e.g.,* key is knowing substitution pattern in mono alphabetic substitution cipher

Q: how do Bob and Alice agree on key value?

Public Key Cryptography

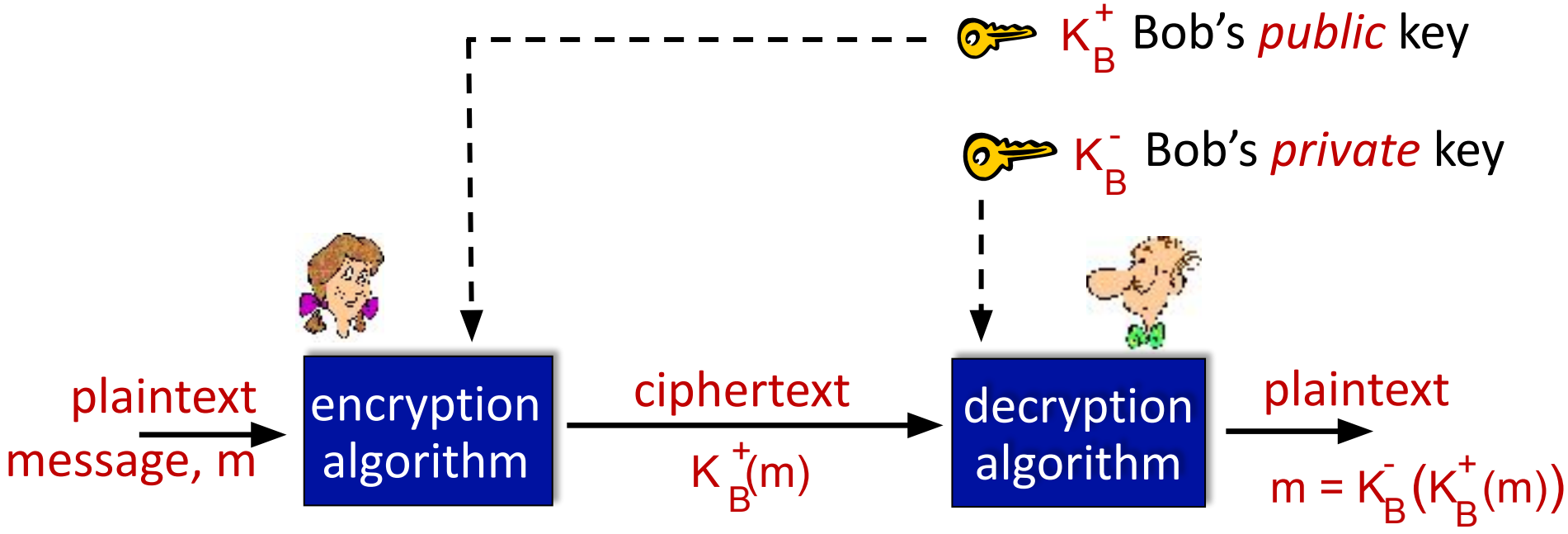
symmetric key crypto:

- requires sender, receiver know shared secret key
- Q: how to agree on key in first place (particularly if never "met")?

public key crypto

- *radically* different approach [Diffie-Hellman76, RSA78]
- sender, receiver do *not* share secret key
- *public* encryption key known to *all*
- *private* decryption key known only to receiver

Public Key Cryptography



Wow - public key cryptography revolutionized 2000-year-old (previously only symmetric key) cryptography!

- similar ideas emerged at roughly same time, independently in US and UK (classified)

Public key encryption algorithms

requirements:

1. need $K_B^+(\cdot)$ and $K_B^-(\cdot)$ such that

$$K_B^-(K_B^+(m)) = m$$

2. given public key K_B^+, it should be impossible to compute private key K_B^-

RSA: Rivest, Shamir, Adelson algorithm

RSA: getting ready

- message: just a bit pattern
- bit pattern can be uniquely represented by an integer number
- thus, encrypting a message is equivalent to encrypting a number

example:

- m= 10010001. This message is uniquely represented by the decimal number 145.
- to encrypt m, we encrypt the corresponding number, which gives a new number (the ciphertext).

RSA: Creating public/private key pair

1. choose two large prime numbers *p, q.* (e.g., 1024 bits each)

2. compute *n = pq, z = (p-1)(q-1)*

3. choose *e (*with *e<n)* that has no common factors with z (*e, z* are "relatively prime").

4. choose *d* such that *ed-1* is exactly divisible by *z*. (in other words: *ed* mod *z = 1*).

5. *public* key is *(n,e)*. *private* key is *(n,d)*.

K_B^+ K_B^-

RSA: encryption, decryption

0. given (n,e) and (n,d) as computed above

1. to encrypt message m ($<n$), compute
   $c = m^e \bmod n$

2. to decrypt received bit pattern, c, compute
   $m = c^d \bmod n$

magic happens! $m = (\underbrace{m^e \bmod n}_{c})^d \bmod n$

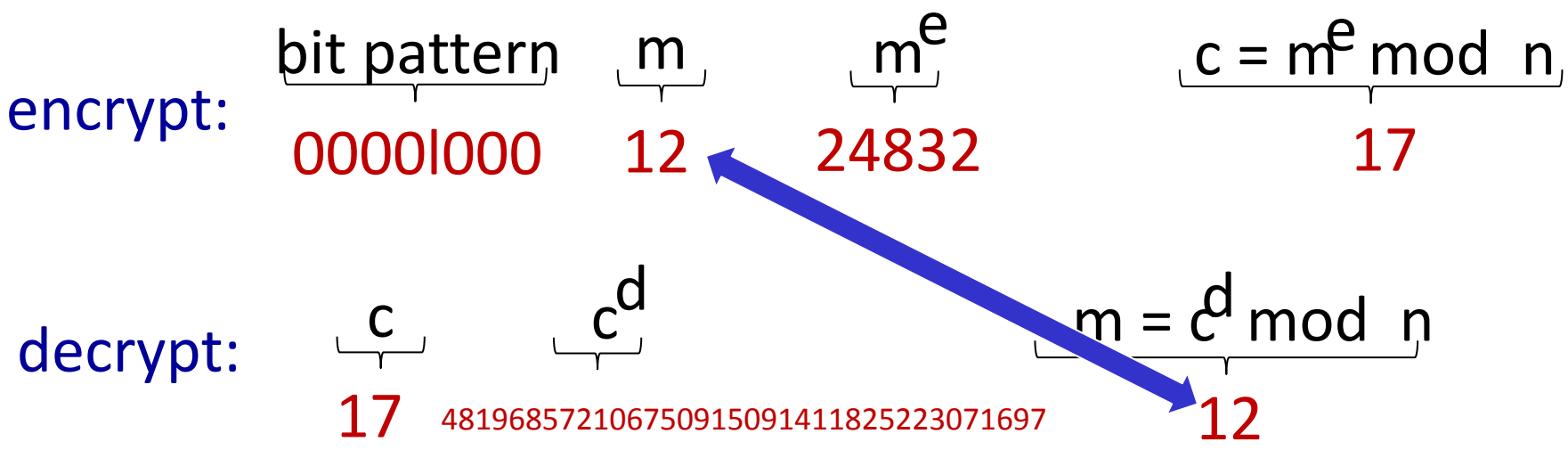
RSA example:

Bob chooses *p=5, q=7*. Then *n=35, z=24*.

e=5 (so *e, z* relatively prime).

d=29 (so *ed-1* exactly divisible by z).

encrypting 8-bit messages.



Q&A